



# LED Sound Meter

Written By: Sam Berg

## PARTS:

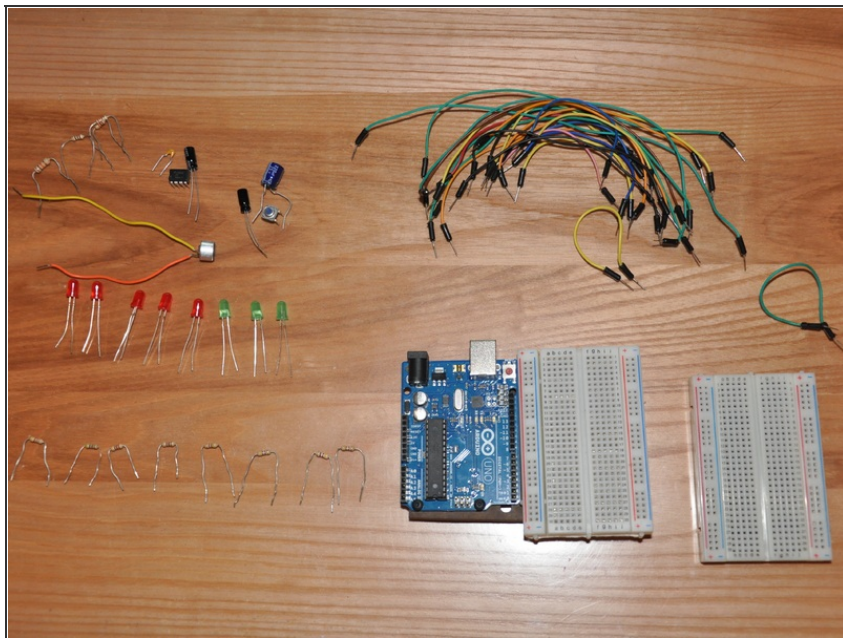
- [Electret Microphone \(1\)](#)
- [LED \(6-10\)](#)
- [LM386 audio op-amp \(1\)](#)  
*[Any audio op-amp will work but you will need to look up the datasheet for yourself.](#)*
- [Arduino Uno \(1\)](#)
- [Small Breadboard \(2\)](#)
- [Jumper Wire \(20-25\)](#)
- [1/4W 470 ohm resistor \(6-10\)](#)
- [Capacitor, 220 F electrolytic \(1\)](#)
- [Electrolytic capacitor, 10uF \(2\)](#)
- [Push-button switch \(1\)](#)
- [Resistor, 10kΩ, 1/4W \(2\)](#)
- [.05μF Ceramic Disc Capacitor \(1\)](#)

## SUMMARY

In this guide you will learn to create an array of LEDs that visually represent the volume picked up by an electret microphone. You will learn how to wire 3 different circuits which comprise the LED array (which is really many individual circuits), the microphone, and a button which can be used to change the display mode of the LEDs. This guide will not cover all of the coding involved in making this, but it WILL give you the full code needed to make this work. The code is thoroughly commented and will teach you a lot if you are interested in

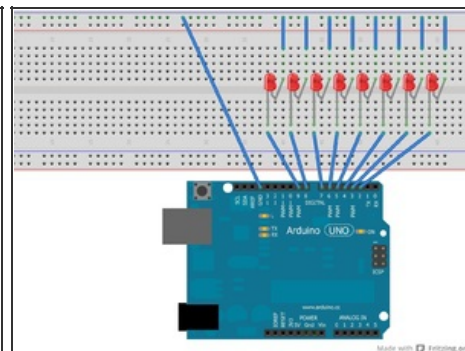
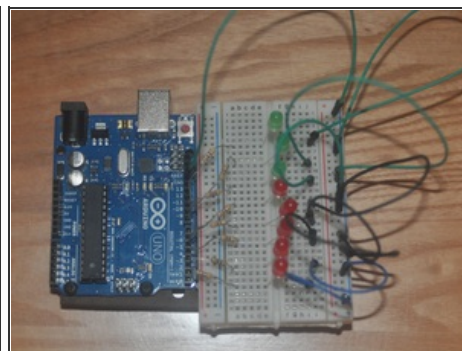
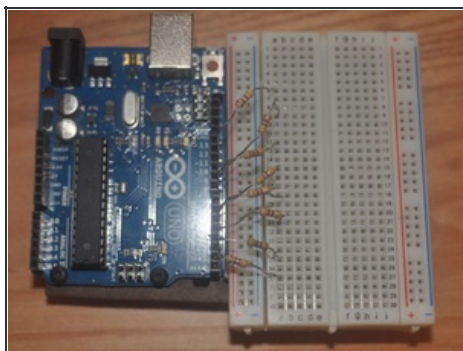
the programming behind this.

## Step 1 — Overview



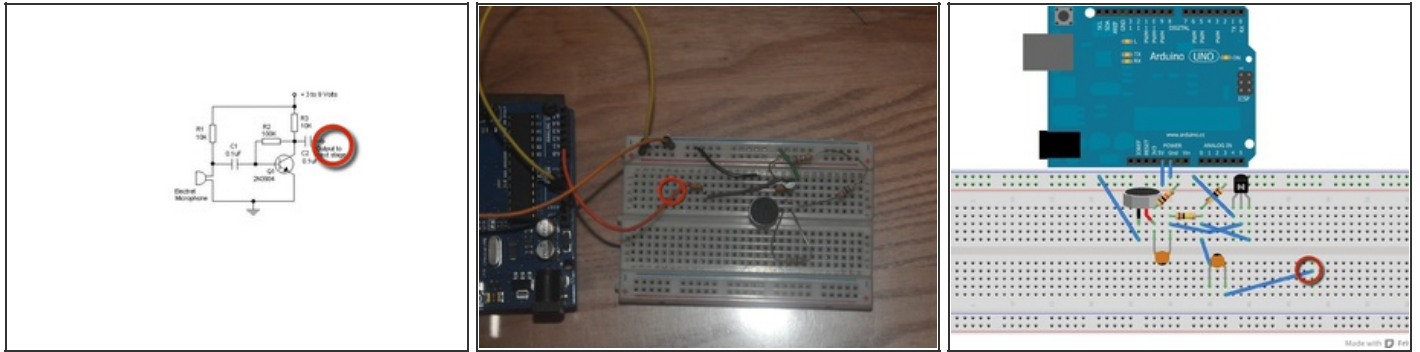
- This project will use an electret microphone to light up an array of LEDs. The parts used are shown in the picture to the left.

## Step 2 — Creating the LED circuit.



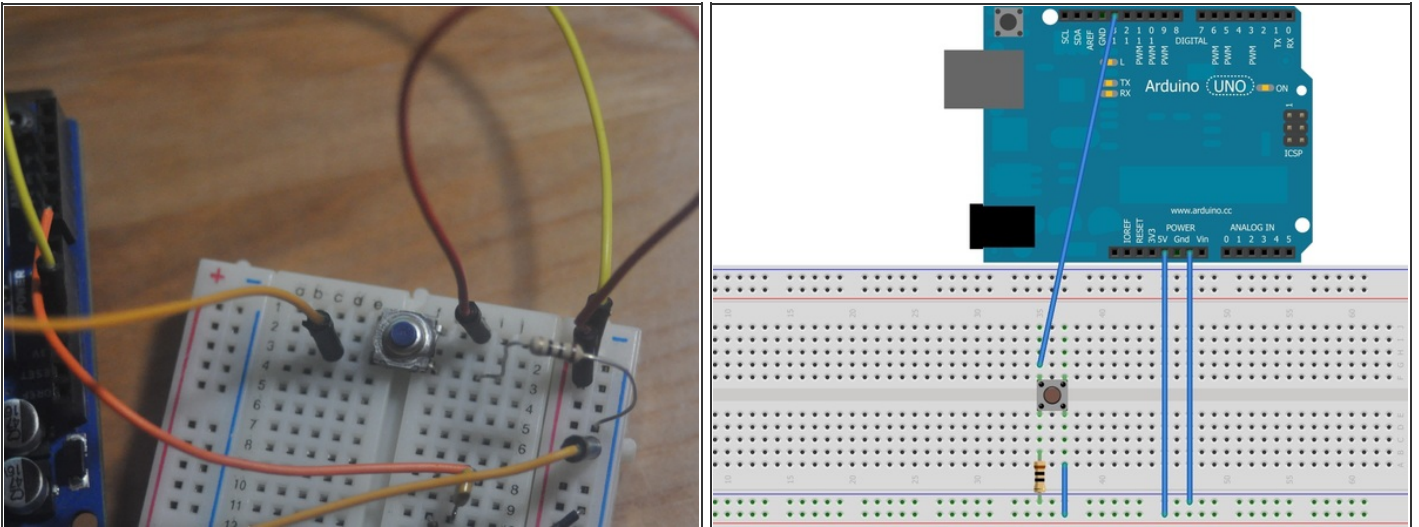
- Connect all of your 470 ohm resistors to consecutive digital pins on your Arduino, and connect the other ends to different rows on your breadboard.
- Next attach the long (positive) leads of your LEDs to the same row as each resistor. Attach the other leads of the LEDs to the corresponding rows on the other side of the board.
- Finally, connect the ground pin from your Arduino to the negative column on your breadboard. Then connect all your LEDs to that negative column.

### Step 3 — Creating the microphone circuit.



- First set up the 5v and ground on your breadboard by hooking up 5v to one of the red "+" columns and ground to one of the blue "-" columns.
- Next connect the negative lead of your microphone to ground.
- Connect the positive lead of your microphone to a 0.1 uF capacitor. We will call this Capacitor 1.
- Next attach a transistor, with the base (middle) pin connected to the other lead of the capacitor from step 3.
- Attach a wire from the collector (right) pin of the transistor to another 0.1 uF capacitor. We'll call this Capacitor 2. Then connect the other end of that capacitor to analog pin 0. You can technically use any analog pin you want but you will have to change the number in your code.
- Now attach the emitter (left) pin of the transistor to ground.
- Now take two 10k resistors and attach one end of each to 5v. Attach one of the resistors to the positive side of the microphone. Attach the other one to the emitter (right) pin of the transistor. You are now done with the microphone circuit. Keep in mind that all the red circles in the pictures represent what you need to attach to analog pin 0.

## Step 4 — Setting up the button.



- First place the button across the middle of the breadboard.
- Attach one of the pins of the button to ground.
- Next attach the pin on the same side of the button, but the opposite side of the breadboard, to digital pin 13. It can be to any digital pin you aren't using but you will have to change that in the code.
- Finally, attach the pin on the same side of the breadboard as the resistor to 5v and you are done setting up the button.
- The button will be used to change display modes.

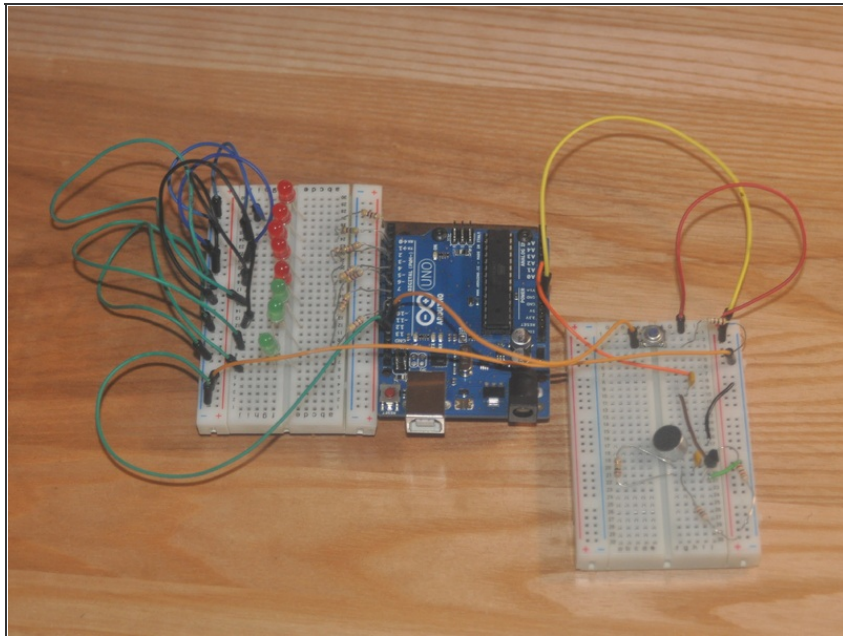
## Step 5 — The Code



- Copy and paste the code from [here](#).
- There are several adjustments you may need to make if you didn't use the same digital and analog pins I used.
  - At the top you will see `int led[] = {2,3,4,5,6,7,8,9,10};`. You need to change these numbers to whatever digital pins you used for your LEDs. It has to be in ascending order because the program uses `for` loops to cycle through the array, so it would light up your LEDs in a random order if they are not listed in ascending order in the array.
  - The same thing goes for the button. At the top it says `int button = 13;`. Change the 13 to whatever digital pin you used.
  - The last thing is `int microphonepin = 0;`. Once again, change that to whatever analog pin you used. Remember that this one has to be an analog pin, otherwise it won't work.



## Step 6 — Congratulations, you have made an LED sound meter!



- Keep in mind this device uses a VERY sloppy method for getting a volume level from the microphone. This results in the LEDs flickering a lot, but it DOES work.
- If you want this to function better you could use the code I already wrote but didn't have time to implement, which takes a moving average of the microphone values to account for the input being a wave. It also gives you the highest value during the window of that average that it takes.
- You will find this code in the main loop of the program.
- Enjoy your LED sound meter!

This document was last generated on 2013-02-16 07:08:03 PM.